

CLAIMS

What is claimed is:

- 1 1. A system comprising:
2 a compiler to compile a computer program , the compiler forming
3 optimized code regions to improve performance of the computer program by
4 optimizing frequently executed code and using compiler transformation and
5 hardware support to handle infrequently executed code;
6 a memory to store the compiler;
7 a central processing unit (CPU) cooperatively connected to the memory to
8 run the compiler and to speculatively execute the optimized code regions;
9 a store buffer connected to the CPU to improve the speed at which the
10 CPU speculatively executes the optimized code regions.
- 1 2. The system of claim 1, wherein the store buffer assists the CPU in
2 speculatively executing the optimized code regions by temporarily storing
3 results produced during the speculative execution of the optimized code regions.
- 1 3. The system of claim 1, wherein the compiler improves performance of
2 the computer program by applying standard available optimizations.
- 1 4. The system of claim 1, wherein the compiler improves performance of
2 the computer program by selecting a seed block, duplicating the seed block, and
3 growing the seed block to form a region.
- 1 5. The system of claim 4, wherein the compiler improves performance of
2 the computer program by trimming blocks near a head block of the region to
3 form a trimmed region having improved scheduling cycles.

1 6. The system of claim 4, wherein the compiler improves performance of
2 the computer program by trimming blocks near a tail block of the region to form
3 a trimmed region having improved scheduling cycles.

1 7. The system of claim 4, wherein the compiler improves performance of
2 the computer program by updating a branch frequency for an original code
3 corresponding to the region to form an improved region.

1 8. A method comprising:
2 compiling a computer program;
3 improving performance of the computer program by optimizing
4 frequently executed code and using compiler transformation and hardware
5 support to handle infrequently executed code;
6 storing temporarily the results produced during execution of formed
7 regions to handle infrequent code being actually executed;
8 committing the results produced when the execution of the region is
9 completed successfully.

1 9. The method of claim 8, wherein the improving performance of the
2 computer program further includes applying standard available optimizations to
3 the formed regions.

1 10. The method of claim 8, wherein the improving performance of the
2 computer program further includes selecting a seed block, duplicating the seed
3 block, and growing the seed block to form a region.

1 11. The method of claim 10, wherein the improving performance of the
2 computer program further includes trimming blocks near a head block of the
3 region to form a trimmed region of having improved scheduling cycles.

1 12. The method of claim 10, wherein the improving performance of the
2 computer program further includes trimming blocks near a tail block of the
3 region to form a trimmed region having improved scheduling cycles.

1 13. The method of claim 10, wherein the improving performance of the
2 computer program further includes updating a branch frequency for an original
3 code corresponding to the region to construct an improved region.

1 14. A machine-readable medium comprising instructions which, when
2 executed by a machine, cause the machine to perform operations comprising:
3 compiling a computer program;
4 improving performance of the computer program by optimizing
5 frequently executed code and using compiler transformation and hardware
6 support to handle infrequently executed code;
7 storing temporarily the results of produced during execution of a region
8 to improve performance of the computer program; and
9 committing the results produced when the execution of the region is
10 completed successfully.

1 15. The machine-readable medium of claim 14, wherein the improving
2 performance of the computer program further includes applying standard
3 available optimizations.

1 16. The machine-readable medium of claim 14, wherein the improving
2 performance of the computer program further includes selecting a seed block,
3 duplicating the seed block, and growing the seed block to form a region.

1 17. The machine-readable medium of claim 16, wherein the improving
2 performance of the computer program further includes trimming blocks near a
3 head block of the region.

1 18. The machine-readable medium of claim 16, wherein the improving
2 performance of the computer program further includes trimming blocks near a
3 tail block of the region.

1 19. The machine-readable medium of claim 16, wherein the improving
2 performance of the computer program further includes updating a branch
3 frequency for a original code corresponding the region.